(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 317 832 B1

(12) **EUROPEAN PATENT SPECIFICATION**

(45) Date of publication and mention of the grant of the patent:
14.12.2016 Bulletin 2016/50

(21) Application number: **01965851.7**

(22) Date of filing: **31.07.2001**

(51) Int Cl.:
H04B 1/707 *(2006.01)* ***G06F 9/48*** *(2006.01)*
H04B 1/7117 *(2011.01)*

(86) International application number:
PCT/US2001/024442

(87) International publication number:
WO 2002/011386 (07.02.2002 Gazette 2002/06)

(54) **METHOD AND APPARATUS FOR TIME-SLICED AND MULTI-THREADED DATA PROCESSING IN A COMMUNICATION SYSTEM**

VERFAHREN UND EINRICHTUNG ZUR ZEITSCHEIBEN- SOWIE VIELFADENDATENVERARBEITUNG IN EINEM VERMITTLUNGSSYSTEM

PROCEDE ET APPAREIL POUR UN TRAITEMENT DE DONNEES A TRANCHE DE TEMPS ET MULTIFILIERE DANS UN SYSTEME DE COMMUNICATION

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE TR

(30) Priority: **31.07.2000 US 222007 P**

(43) Date of publication of application:
11.06.2003 Bulletin 2003/24

(73) Proprietor: **Intel Deutschland GmbH**
85579 Neubiberg (DE)

(72) Inventors:
- **RIEKEN, Keith**
  Cupertino, CA 95014 (US)
- **MEDLOCK, Joel, D.**
  Los Gatos, CA 95032 (US)
- **HOLMES, David, M.**
  Cupertino, CA 95014 (US)

(74) Representative: **Goddar, Heinz J.**
Boehmert & Boehmert
Anwaltspartnerschaft mbB
Patentanwälte Rechtsanwälte
Pettenkoferstrasse 20-22
80336 München (DE)

(56) References cited:
WO-A-00/69084 **WO-A-00/69192**
US-A- 6 005 856 **US-A- 6 091 762**

- **RABAEY J M ET AL: "Heterogeneous reconfigurable systems" SIGNAL PROCESSING SYSTEMS, 1997. SIPS 97 - DESIGN AND IMPLEMENTATION., 1997 IEEE WORKSHOP ON LEICESTER, UK 3-5 NOV. 1997, NEW YORK, NY, USA,IEEE, US, 3 November 1997 (1997-11-03), pages 24-34, XP010249763 ISBN: 978-0-7803-3806-7**
- **ABNOUS A ET AL: "Ultra-low-power domain-specific multimedia processors" VLSI SIGNAL PROCESSING, IX, 1996., WORKSHOP ON SAN FRANCISCO, CA, USA 30 OCT.-1 NOV. 1996, NEW YORK, NY, USA,IEEE, US, 30 October 1996 (1996-10-30), pages 461-470, XP010199037 ISBN: 978-0-7803-3134-1**
- **RABAEY J M: "Reconfigurable processing: the solution to low-power programmable DSP" ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, 1997. ICASSP-97., 1997 IEEE INTERNATIONAL CONFERENCE ON MUNICH, GERMANY 21-24 APRIL 1997, LOS ALAMITOS, CA, USA,IEEE COMPUT. SOC, US, vol. 1, 21 April 1997 (1997-04-21), pages 275-278, XP010226188 ISBN: 978-0-8186-7919-3**

Note: Within nine months of the publication of the mention of the grant of the European patent in the European Patent Bulletin, any person may give notice to the European Patent Office of opposition to that patent, in accordance with the Implementing Regulations. Notice of opposition shall not be deemed to have been filed until the opposition fee has been paid. (Art. 99(1) European Patent Convention).

Printed by Jouve, 75001 PARIS (FR)

(Cont. next page)

- TUTTLEBEE W: "Software radio-impacts and implications" SPREAD SPECTRUM TECHNIQUES AND APPLICATIONS, 1998. PROCEEDINGS., 1998 IEEE 5TH INTERNATIONAL SYMPOSIUM ON SUN CITY, SOUTH AFRICA 2-4 SEPT. 1998, NEW YORK, NY, USA,IEEE, US, vol. 2, 2 September 1998 (1998-09-02), pages 541-545, XP010307620 ISBN: 978-0-7803-4281-1
- ARRIGO J F ET AL: "Adaptive FEC on a reconfigurable processor for wireless multimedia communications" CIRCUITS AND SYSTEMS, 1998. ISCAS '98. PROCEEDINGS OF THE 1998 IEEE IN TERNATIONAL SYMPOSIUM ON MONTEREY, CA, USA 31 MAY-3 JUNE 1998, NEW YORK, NY, USA,IEEE, US, vol. 4, 31 May 1998 (1998-05-31), pages 417-420, XP010289453 ISBN: 978-0-7803-4455-6
- HALTER S ET AL: "Reconfigurable signal processor for channel coding and decoding in low SNR wireless communications" SIGNAL PROCESSING SYSTEMS, 1998. SIPS 98. 1998 IEEE WORKSHOP ON CAMBRIDGE, MA, USA 8-10 OCT. 1998, NEW YORK, NY, USA,IEEE, US, 8 October 1998 (1998-10-08), pages 260-274, XP010303741 ISBN: 978-0-7803-4997-1

Description

PRIORITY INFORMATION

[0001] This application claims priority from the Provisional Application entitled "Apparatus and Method for Despreading Data in a CDMA System", U.S. Serial No. 60/222,007, filed on July 31,2000.

CROSS REFERENCE TO RELATED APPLICATION

[0002] Related applications are:

"Generic Finger Architecture for Spread Spectrum Applications", filed concurrently herewith;
"Apparatus and Methods for Sample Selection and Reuse of Rake Fingers in Spread Spectrum Systems", filed concurrently herewith; and
"Apparatus and Method for Configurable Multi-dwell Search Engine for Spread Spectrum Applications", filed concurrently herewith.

BACKGROUND OF THE INVENTION

[0003] This invention relates generally to wireless communication systems.

[0004] Wireless communication has extensive applications in consumer and business markets. Among the many communication applications/systems are: mobile wireless, fixed wireless, unlicensed Federal Communications Commission (FCC) wireless, local area network (LAN), cordless telephony, personal base station, telemetry, and others.

[0005] Signal processing protocols and standards have proliferated with advances in wireless communications devices and services. Current communications protocols include Frequency Division Multiplexing (FDM), Time Division Multiple Access (TDMA), and Code Division Multiple Access (CDMA). The United States, Europe, Japan, and Korea have all developed their own standards for each communications protocol. TDMA standards include Interim Standard-136 (IS-136), Global System for Mobile (GSM), and General Packet Radio Service (GPRS). CDMA standards include Global Positioning System (GPS), Interim Standard-95 (IS-95) and Wide Band CDMA (WCDMA). Wireless communications services include paging, voice and data applications.

[0006] In many cases, within the same field of applications, different systems use incompatible modulation techniques and protocols. Consequently, each system may require unique hardware, software, and methodologies for baseband processing. This practice can be costly in terms of design, testing, manufacturing, and infrastructure resources. As a result, a need arises to overcome the limitations associated with the varied hardware, software, and methodology of processing digital signals in each of the varied applications.

[0007] Until recently, individual wireless communications devices supported a single communications standard. In theory, however, a wireless communications device can be designed using a general purpose Digital Signal Processor (DSP) that is programmed first to realize a first set of functional blocks specifying the minimum performance requirements for a first application and can be reprogrammed to realize a second set of functional blocks to provide a second application. To achieve these minimum performance requirements, system designers design algorithms (sequences of arithmetic, trigonometric, logic, control, memory access, indexing operations, and the like) to encode, transmit, and decode signals. These algorithms are typically specified in software. The set of algorithms which achieve the target performance specification is collectively referred to as the executable specification. This executable specification can then be compiled and run on the DSP, typically via the use of a compiler. Despite the increasing computational power and speed of general purpose DSPs and decreasing memory cost and size, designers have not been able to satisfy cost, power and speed requirements simply by programming a general purpose DSP with the executable specification for a standard-specific application.

[0008] Additional dedicated high-speed processing is required, a need which has traditionally been met using an application-specific processor. As used herein, an application-specific processor is a processor that excels in the efficient execution (power, area, flexibility) of a set of algorithms tailored to the application. An application-specific processor, however, fares extremely poorly for algorithms outside the intended application space. In other words, the improved speed and power efficiency of application-specific-processors comes at the cost of function flexibility.

[0009] Demand is now growing for wireless communications devices that support multiple applications and varying grades of services over multiple standards. In particular, demand is growing for cellular handsets, which are one type of wireless communications device, to support multiple applications and services over multiple standards. Today's solution to this problem is to essentially connect multiple application-specific processors together to obtain multi-standard operation, thereby adding cost in terms of design resources, design time, and silicon area.

[0010] Cellular handsets and basestations, including PCS (Personal Communications Services) and 3-G (Third Generation) devices, need to acquire certain cell specific information and characteristics before negotiating a service with a base station. For this purpose, each base station transmits certain cell specific information necessary for a user to acquire services such as paging or cellular telephony from the base station.. For example, in CDMA based systems, the cell specific information is contained in pilot and/or synchronization channels. The pilot and/or synchronization channels are spread and scrambled with cell specific pseudo-random noise (PN) or gold code sequences. At the receiver, the scrambled

sequence is converted back to the original data sequence.

[0011] Multiple users are typically served at a single base station. In CDMA systems, each user is assigned an orthogonal code from a set of orthogonal codes and data that is transmitted from the base station to the user is spread according to the assigned orthogonal code. Even though users operate on the same frequency at the same time, the use of orthogonal codes allow multiple users to be distinguished from one another.

[0012] Some data processing systems employ a generic time-sliced architecture to perform data processing functions. Typically, a user builds an application on top of a generic time-sliced architecture based on fixed constraints inherent in the generic time-sliced architecture. For example, data processing engines are designed to optimize the performance on the silicon process for a generic set of operations. When using a generic time-sliced architecture, a user designing an application has the responsibility of real time scheduling (e.g., reading and writing to and from memory) on the generic time sliced architecture. This responsibility is particularly burdensome if a high volume of data comes in at a very high speed, such as data arriving in wireless communications. In addition, even if a user is able to write applications that successfully schedule real time processes, the user still has the burden of managing and maintaining real time aspects of the processing at the lowest level (i.e., below radio frame).

[0013] Rabaey J. M. et al. "Heterogeneous reconfigurable systems" Signal Processing Systems, 1997. SIPS 97 - Design and Implementation., 1997, IEEE Workshop on Leicester, UK 3-5 Nov. 1997, New York, NY, USA, IEEE, US, 3 November 1997 (1997-11-03), pages 24-34, disclose a heterogeneous reconfigurable architecture, in which a communication network is provided between a control processor and a plurality of arithmetic processors. It is generally indicated that such an approach has the advantage that it can exploit the right levels of granularity and flexibility as needed by the application domain, yet that it can be supported by a well-defined design and implementation methodology.

[0014] Abnous A et al. "Ultra-low-power domain-specific multimedia processors" VLSI Signal Processing, IX, 1996., Workshop on San Francisco, CA, USA, 30 Oct. - 1 Nov. 1996, New York, NY, USA, IEEE, US, 30 October 1996 (1996-10-30), pages 461-470, XP010199037 discloses an architecture template comprising a control processor, a communication network and a plurality of satellite processors. The communication network is reconfigured by the control processor such that all temporal correlations in a given stream of data are preserved, and the amount of switching activity is minimized.

[0015] WO 00/69084 representing prior art according to Article 54 (3) EPC discloses a digital wireless communication device comprising a software-programmable processor, a heterogeneous reconfigurable multiprocessing logic circuit and a bus connecting the software-programmable processor and the heterogeneous reconfigurable multiprocessing logic circuit. The heterogeneous reconfigurable multiprocessing logic circuit comprises a set of heterogeneous signal processing kernels and a reconfigurable data router interconnecting the heterogeneous signal processing kernels. Accordingly, a single product platform can be reconfigured for multiple standards, applications and services.

[0016] In view of the foregoing, it is desirable to provide a specific processor that supports disparate communications and signal processing standards in a cost, area, and power efficient fashion. It is further desirable to provide a method and apparatus that automates time scheduling aspects of data processing by optimizing a specific time-sliced and multi-threaded architecture in a communication system.

SUMMARY OF THE INVENTION

[0017] This invention provides processor architectures that enable high throughput chip rate processing. In an exemplary embodiment, parallel processing techniques and control structures are used to provide flexibility in managing buffer and processing requirements of high performance spread spectrum systems. An architecture in accordance with an exemplary embodiment provides optimization of buffer and processing requirements in a highly flexible micro-architectural implementation. Advantages of implementing the micro-architectures in accordance with embodiments of this invention include: (1) maximizing the efficiency of processing by scaling throughput relative to input data rate; (2) increasing flexibility across a wide range of searching/tracking configurations; (3) improving scalability across variable data rates associated with users; (4) providing software control of finger scheduling to accommodate varying requirements; and (5) providing search control flexibility.

[0018] In an exemplary embodiment, a time-sliced and multi-threaded architecture is designed by conducting a thorough analysis of a range of applications and building a specific processor to accommodate the range of applications. In one embodiment, the thorough analysis includes extracting real time aspects from each application, determining optimal granularity in the architecture based on the real time aspects of each application, and adjusting the optimal granularity based on acceptable context switching overhead.

BRIEF DESCRIPTION OF THE DRAWINGS

[0019]

FIGURE 1 is a block diagram of an exemplary multi-threaded architecture for spread spectrum chip rate processing in accordance with an embodiment of the invention.

FIGURE 2 is a flow chart of an optimal architecture implementation process in accordance with an em-

bodiment of the invention.

DETAILED DESCRIPTION OF THE INVENTION

[0020] Figure 1 illustrates a basic multi-threaded micro-architecture 100 in accordance with an exemplary embodiment of the invention. The multi-threaded micro-architecture 100 can be leveraged across numerous multi-user spread spectrum receiver applications. The multi-threaded micro-architecture 100 includes a data cache 102, a first finger processing element 104, a second finger processing element 106, an "nth" (where "n" represents an arbitrary, configurable number) finger processing element 108, and a master control unit 110. The master control unit includes a time slot table 112 and a partial sums search table 114. In an exemplary embodiment, the number of finger processing elements in the architecture 100 is dependent on various design constraints and can vary from architecture to architecture without departing from the essence of this invention. For ease of explanation, three finger processing elements 104, 106, 108 are illustrated in Figure 1. Each finger processing element includes a secondary cache 122, a data selection module 124, a despread/dechannelize datapath 126, and a symbol integration module 128.

[0021] Incoming digital data, which contains code modulated user information, is buffered in the data cache 102. The data cache 102 is shared by all finger processing elements 104-108. Each finger processing element 104-108 contains the necessary datapath for despreading, dechannelization, and symbol integration of the individual user channels. The master control unit 110 allocates time slots, maintains synchronization of the finger processing elements 104-108, and maximizes throughput. For example, the partial sums search table 114 is allocated on a per searcher basis to extend search control flexibility across time slots. In an exemplary embodiment, the master control unit 110 is linked to an external processing element to manage time slot allocation among finger processing elements 104-108.

[0022] In an exemplary embodiment, the data cache 102 is a parallel port memory that is configured to enable multi-threaded access at virtually the same time. In one embodiment, a hierarchical caching structure is implemented where the data cache 102 includes a primary cache that is accessible by each finger processing element 104-108 in a round-robin manner. Each finger processing element 104-108 includes a secondary cache that is configured to prefetch data from the primary cache and store such prefetched data. For example, if there are 16 finger processing elements, the first finger processing element 104, during its turn to access the primary cache, prefetches 16 samples, such that it has 16 clock cycles of time before it needs to prefetch again. Similarly, during the next clock cycle, the second finger processing element 106 prefetches 16 samples and so on. This way, the data cache 102 can be built as a multi-ported RAM (e.g., 16-ported) at very low cost. Further details on caching systems are disclosed in the above-referenced, concurrently filed application entitled "Generic Finger Architecture for Spread Spectrum Applications".

[0023] The processor architectures in accordance with various embodiments of the present invention use a common processing element (e.g., finger processing elements 104-108) to support varying spreading factors, modulation schemes, and user data rates. Furthermore, the processor architecture enables flexible searching algorithms with variable length search windows, which are made possible in part by a shared search table and a master control. In addition, multiple data stream selection, such as varying antenna configuration, can be used to further reduce silicon costs for manufacturing the finger processing elements. Further details on appropriate processor architecture are disclosed in the above-referenced, concurrently filed application entitled: "Apparatus and Methods for Sample Selection and Reuse of Rake Fingers in Spread Spectrum Systems."

[0024] In one embodiment, the multi-threaded micro-architecture 100 is a hardware computation resource that can be applied to a single computation process (e.g., a multipath of a given channel). In another embodiment, the computation resource provided by the multi-threaded micro-architecture 100 can be enhanced by running the multi-threaded micro-architecture 100 at a clock rate higher than that required by a process (e.g., higher than the data rate for a communication protocol). In this manner, resources of individual computation components, such as the multi-threaded micro-architecture 100, can be time-shared across multiple computation processes (e.g., several multipaths and/or multiple channels). Additional information on the design and implementation of configurations into a configurable communication device is provided in a co-pending application bearing serial number 09/492,634, entitled "Improved Apparatus and Method for Multi-Threaded Signal Processing."

[0025] Figure 2 illustrates an exemplary process for designing an optimal time-sliced and multi-threaded architecture. At step 202, symbol processing requirements are determined. In an exemplary embodiment, a microprocessor workstation receives inputs of a range of applications to be supported by the architecture being designed.

[0026] The process of determining an optimal component combination that maximizes the efficiency of the multi-threaded chip rate processor involves consideration of various system requirements. In an exemplary embodiment, system requirements include: (1) possible antenna configurations, incoming data rates, and combining requirements; (2) downstream processing requirements that dictate output symbol rate requirements; (3) processor interface requirements that impact the efficient allocation of finger processing elements; (4) variations in the spreading/modulation processes that are applied to the expected data streams; and (5) environmental requirements, such as search time, simultaneous multipath tracking, and peak/average channel capacity re-

quirements. In an exemplary embodiment, after consideration of system requirements, including the ones listed above, key architecture aspects can be determined. Examples of the key aspects include: data cache memory requirements, number of finger processing elements, performance requirements of the finger processing elements, performance constraints of the finger processing elements, memory bandwidth requirements of the data cache, and time slot size to accommodate convenient downstream processing.

[0027] In an exemplary embodiment, fundamental processing units are defined by applying a profiling process. The fundamental processing units are parameterizable processing blocks that may be application specific but can be enabled for a variety of protocols. The profiling process is performed from a system and hardware perspective to optimize time sliced and multi-threaded architecture. Illustrative examples of fundamental processing units are the hardware kernels described in Figure 2 of co-pending U.S. Application Serial No. 09/772,584 entitled "A Wireless Spread Spectrum Communication Platform Using Dynamically Reconfigurable Logic." Additional information on the profiling process is provided in co-pending U.S. application serial no. 09/565,654, entitled "Method of Profiling Disparate Communications and Signal Processing Standards and Services."

[0028] During profiling, a determination is made of the lowest level of timing granularity needed. In digital signal processing the fundamental time unit is ordinarily the over-sampling rate of the originally transmitted signal which typically is the Nyquest rate. In a typical spread spectrum system, the fundamental unit of time is the chip rate. The fineness of a desired granularity is determined by profiling the types of processing required for each application. Further, in determining granularity, a trade off between fine granularity and high context switching overhead should be considered. In general, the finer the granularity, the better the algorithmic performance. But at the same time, the finer the granularity, the more context switching is required in hardware. In a preferred embodiment, the granularity should be fine enough that the targeted algorithms perform signal processing efficiently while allowing a given process of the targeted algorithms to run in the processor for as long as possible, thus, minimizing context switching overhead.

[0029] In an exemplary embodiment, the time-sliced architecture in accordance with the invention is capable of supporting multiple spread spectrum applications that run at different granularities when optimized. For example, a first application may be optimized at 8x chip rate granularity while a second application may be optimized at 1x chip rate granularity.

[0030] In another exemplary embodiment, the time-sliced architecture is able to call programming across different protocols in a given application space. In contrast to prior art architectures where the overall concern is regarding hardware resource utilization at a known and fixed performance level, the architecture in accordance with embodiments of this invention is not only application specific (for a set of applications) but also flexibly reconfigurable to support multiple applications. In one embodiment, the present architecture enables speed grading (i.e., sorting and assembly of components into useable devices in accordance with their demonstrated operating speed instead rejection of components for failure to meet a specified operating speed) to control available flexibility. That is, the architecture can be configured into different channel densities depending on the number of logical processors it supports for each application.

[0031] At step 204, the target silicon processes needed to achieve the fundamental processing units defined in the previous step (i.e., profiling) are determined. That is, actual physical parts that are capable of delivering each type of processes are determined. For example, most communication operations are linear, so adder and multiplier processing units are frequently required. Thus, during this step, for a given application, the physical location of each necessary adder and/or multiplier (as well as the physical locations of other processing units) on silicon is determined based on data control flow and input/output location.

[0032] At step 206, the input and output data rates are determined for each application. In an exemplary embodiment, the input data rate is calculated on a data-samples-per-second-provided-at-input basis. Output are determined by the worst case minimum rate reduction that occurs in the signal processing path.

[0033] At step 208, the size of the data cache 102 is determined. The appropriate size for data cache 102 for a spread spectrum application is determined based on balancing a trade-off between the size of the implementation (in terms of actual die size) and the delay spread that is associated with the mobile terminals or handsets. Typically, all mobile terminals in the spread spectrum system are operating in the same frequency range.Thus, the data cache 102 should be able to support two or more mobile terminals simultaneously at any given time. In an exemplary embodiment, a parallel port memory is used as the data cache 102 and a hierarchical caching structure that allows multiple threads to access the same data at the same time is implemented. In the hierarchical caching structure, a secondary cache associated with each processing thread prefetches data from a primary cache for that processing thread.

[0034] At step 210, a sensitivity analysis is performed. That is, varied combinations of time slot sizes and processing threads are checked for an optimized combination. For example, the optimal trade-off between context switching overhead and the size of the processing granularity is determined. In an exemplary embodiment, varying time slot sizes, finger processing element numbers, and independent data cache read ports are tested. The optimal number and size are determined in accordance with optimizing the complexity of silicon, including size, and channel capacity requirements.

[0035] Variability in time scheduling is determined

based on basic time units. In other words, once basic time units have been determined, then variability in scheduling (e.g., timing of the occurrence of certain processes, number of each process per algorithm, etc.) for each algorithm is determined. For example, a given logic algorithm may require use of multiple processing threads. Thus, an optimal trade-off between the number of logic algorithms running on the system and the amount of time needed to run each algorithm should be determined in view of the overall goal of maximizing channel density.

[0036] In an exemplary embodiment, real time scaling can be achieved. For example, during off-peak hours, some or all logical threads may be disabled to conserve power consumption.

[0037] The foregoing examples illustrate certain exemplary embodiments of the invention from which other embodiments, variations, and modifications will be apparent to those skilled in the art. The invention should therefore not be limited to the particular embodiments discussed above, but rather is defined by the claims.

Claims

1. A method for building a time-sliced and multi-threaded architecture comprising resources of individual computation components time-shared across multiple computation processes in a spread spectrum system, comprising the steps of :

   (a) analyzing a set of spread spectrum applications, said analyzing including the steps of:

   (i) extracting real time aspects including input data rates and output data rates for each application in said set of applications, said extracting including profiling fundamental processing elements in each application in said set of applications;
   (ii) determining an optimal granularity based on said real time aspects; and
   (iii) adjusting said optimal granularity based on a context switching overhead; and

   (b) building a specific time-sliced and multi-threaded architecture capable of supporting said range of applications based on said analyzing,
   wherein said building includes the steps of:

   determining a size for a data cache based on said extracting;
   implementing a hierarchical caching structure in said data cache; and
   applying said data cache in said specific time-sliced architecture.

2. The method of claim 1, wherein said determining includes the step of:

   determining a lowest level of granularity needed for each application in said set of applications.

3. The method of claim 1, wherein said adjusting includes the step of:

   performing a sensitivity analysis.

4. The method of claim 3, wherein said performing includes the step of:

   determining an optimal trade-off between said context switching overhead and said optimal granularity.

5. A computer program product comprising logic code that, when executed on a logic, implements the method according to one of claims 1 to 4.

Patentansprüche

1. Verfahren zum Aufbauen einer Zeitscheiben- und Multithread-Architektur, die Ressourcen einzelner Berechnungskomponenten umfasst, die zeitlich über mehrere Berechnungsprozesse in einem Spreizspektrumsystem geteilt werden, das die folgenden Schritte umfasst:

   (a) Analysieren einer Menge von Spreizspektrumanwendungen, wobei das Analysieren die folgenden Schritte beinhaltet:

   (i) Extrahieren von Echtzeitaspekten einschließlich Eingangsdatenraten und Ausgangsdatenraten für jede Anwendung in der Menge von Anwendungen, wobei das Extrahieren Profilieren grundlegender Verarbeitungselemente in jeder Anwendung in der Menge von Anwendungen beinhaltet;
   (ii) Bestimmen einer optimalen Granularität basierend auf den Echtzeitaspekten; und
   (iii) Anpassen der optimalen Granularität basierend auf einem Kontextwechseloverhead; und

   (b) Aufbauen einer speziellen Zeitscheiben- und Multithread-Architektur, die in der Lage ist, den Bereich von Anwendungen zu unterstützen, basierend auf dem Analysieren, wobei das Aufbauen die folgenden Schritte beinhaltet:

   Bestimmen einer Größe für einen Datencache basierend auf dem Extrahieren;
   Implementieren einer hierarchischen Cachestruktur in dem Datencache; und

Anwenden des Datencache in der speziellen Zeitscheibenarchitektur.

2. Verfahren nach Anspruch 1, wobei das Bestimmen den folgenden Schritt beinhaltet:

   Bestimmen eines niedrigsten Niveaus einer für jede Anwendung in der Menge von Anwendungen benötigten Granularität.

3. Verfahren nach Anspruch 1, wobei das Anpassen den folgenden Schritt beinhaltet:

   Durchführen einer Empfindlichkeitsanalyse.

4. Verfahren nach Anspruch 3, wobei das Durchführen den folgenden Schritt beinhaltet:

   Bestimmen eines optimalen Kompromisses zwischen dem Kontextwechseloverhead und der optimalen Granularität.

5. Computerprogrammprodukt, das einen Logikcode umfasst, der, wenn er auf einer Logik ausgeführt wird, das Verfahren nach einem der Ansprüche 1 bis 4 implementiert.

Revendications

1. Procédé de création d'une architecture à découpage temporel et multifilière comprenant des ressources de composants de calcul individuels découpés temporellement dans plusieurs processus de calcul dans un système à spectre étalé, comprenant les étapes consistant à :

   (a) analyser un ensemble d'applications à spectre étalé, ladite analyse comprenant les étapes consistant à :

      (i) extraire des aspects en temps réel comprenant des débits de données d'entrée et des débits de données de sortie pour chaque application dans ledit ensemble d'applications, ladite extraction consistant à profiler des éléments de traitement fondamentaux dans chaque application dudit ensemble d'applications ;
      (ii) déterminer une granularité optimale sur la base desdits aspects en temps réel ; et
      (iii) régler ladite granularité optimale sur la base d'un surdébit de changement de contexte ; et

   (b) créer une architecture spécifique à découpage temporel et multifilière capable de prendre en charge ladite gamme d'applications sur la base de ladite analyse,

   ladite création comprenant les étapes consistant à :

      déterminer une taille pour une mémoire cache de données sur la base de ladite extraction ; mettre en oeuvre une structure de mise en cache hiérarchique dans ladite mémoire cache de données ; et appliquer ladite mémoire cache de données dans ladite architecture spécifique à découpage temporel.

2. Procédé selon la revendication 1, dans lequel ladite détermination comprend l'étape consistant à :

   déterminer un niveau minimum de granularité nécessaire pour chaque application dudit ensemble d'applications.

3. Procédé selon la revendication 1, dans lequel ledit réglage comprend l'étape consistant à :

   réaliser une analyse de sensibilité.

4. Procédé selon la revendication 3, dans lequel ladite réalisation comprend l'étape consistant à :

   déterminer un compromis optimal entre ledit surdébit de changement de contexte et ladite granularité optimale.

5. Produit de programme informatique comprenant un code logique qui, lorsqu'il est exécuté sur une logique, met en oeuvre le procédé selon l'une quelconque des revendications 1 à 4.

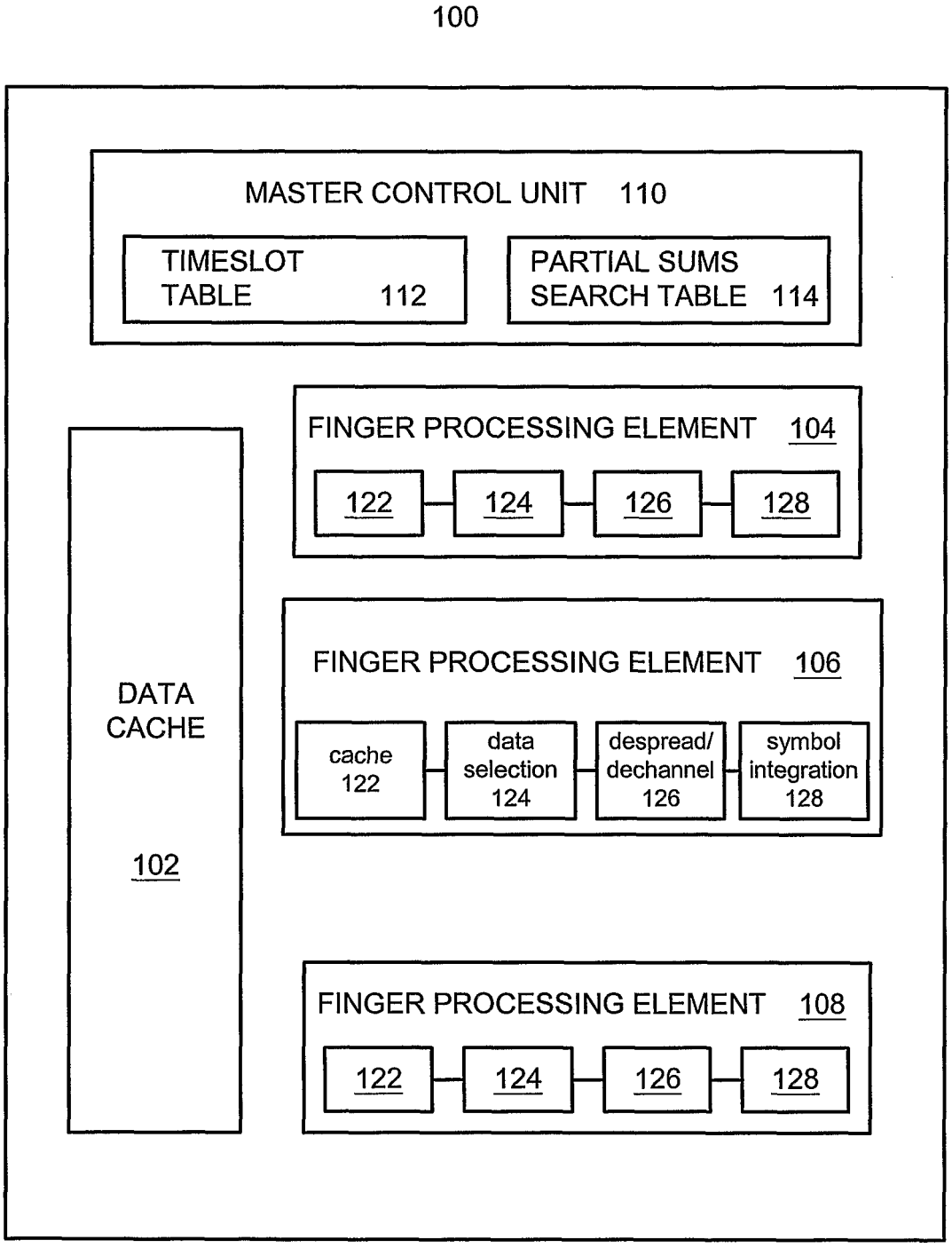

100
MASTER CONTROL UNIT 110
TIMESLOT TABLE 112
PARTIAL SUMS SEARCH TABLE 114
FINGER PROCESSING ELEMENT 104
122
124
126
128
FINGER PROCESSING ELEMENT 106
cache 122
data selection 124
despread/ dechannel 126
symbol integration 128
DATA CACHE
102
FINGER PROCESSING ELEMENT 108
122
124
126
128

Figure 1

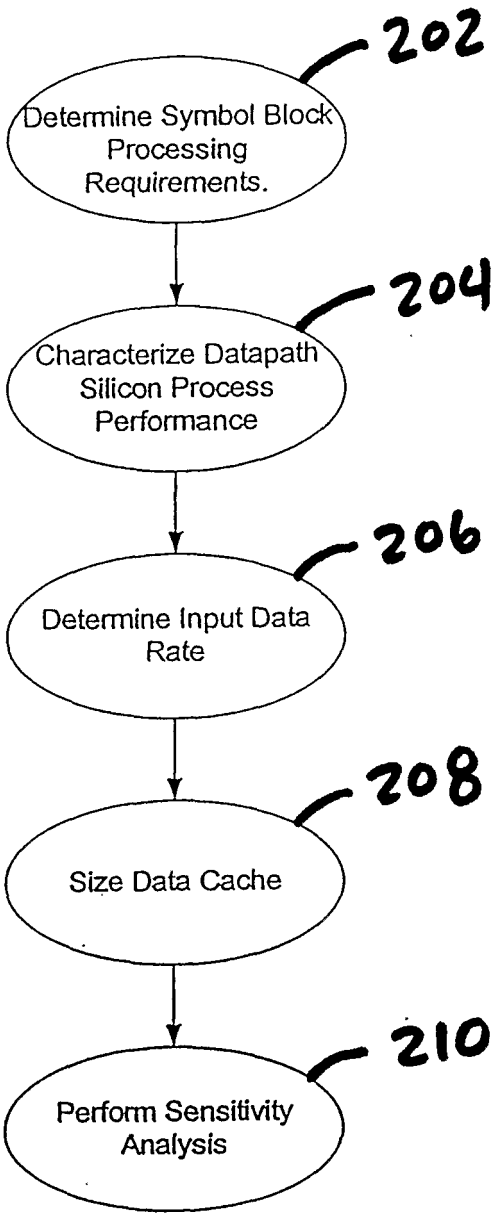

202
Determine Symbol Block Processing Requirements.
204
Characterize Datapath Silicon Process Performance
206
Determine Input Data Rate
208
Size Data Cache
210
Perform Sensitivity Analysis

FIGURE 2

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US 60222007 B **[0001]**
- WO 0069084 A **[0015]**
- US 09772584 B **[0027]**
- US 09565654 B **[0027]**

Non-patent literature cited in the description

- Heterogeneous reconfigurable systems. **RABAEY J. M. et al.** Signal Processing Systems, 1997. SIPS 97 - Design and Implementation., 1997, IEEE Workshop on Leicester. IEEE, 03 November 1997, 24-34 **[0013]**
- Ultra-low-power domain-specific multimedia processors. **ABNOUS A et al.** VLSI Signal Processing, IX, 1996., Workshop on San Francisco. IEEE, 30 October 1996, 461-470 **[0014]**